REMARKS

Claims 1-20 stand rejected as unpatentable over *Schneck* (US 5,933,498) in view of newly-cited *Alexander* (US 6,134,593). The Applicants respectfully traverse this rejection.

Independent Claim 1 defines a method for restricting installation of a software product onto a local machine. This method includes the step of "generating an installer identifier based on the software product, in response to a request to install the software product on the local machine". That installer identifier then is compared to a stored installer identifier on the software product, to verify that the license of usability with the software is in fact restricted to enabling the installation and execution of a particular software product.

The rejection states that *Schneck* discloses the above-quoted step in Figs. 1-3 and associated text, specifying several items identified in Fig. 3. However, those portions of *Schneck* do not disclose or discuss generating an installer identifier in response to a request to install a software product on a local machine. Instead, those identified portions of *Schneck* pertain to the so-called authoring mechanism, discussed commencing at column 11, line 44 and extending through column 15, line 18. That discussion includes details of the authoring mechanism, including the flowchart on Fig. 7 and a discussion of rule-encrypting steps in column 14. The result, according to *Schneck*, is a package of data that may be provided to a user in certain ways, column 15, lines 8-18. However, nothing in that passage or elsewhere in *Schneck* discloses the step of "generating an installer identifier based on the software product, in response to a request to install the software product on a local machine.

This step in the Applicants' claimed method enables the following step in Claim 1, namely, comparing the generated installer identifier. Because *Schneck* does not disclose the step of generating an installer identifier in response to a request to install the software product, it follows that *Schneck* also does not compare the generated installer identifier to a stored installer identifier as required in the method of Claim 1. Figs. 10(a-b) and columns 17-20 of *Schneck* are identified, in the rejection, as teaching the Applicants' step of "comparing the generated installer identifier to a stored installer identifier". Although Figs. 10(a-b) do disclose data access according to *Schneck*, neither those figures nor the related discussion commencing in column 17 disclose comparing a generated installer identifier, in response to a request to install the software product on the local machine, to a stored installer identifier.

Claim 1 also defines the step of "storing a license file on the local machine..." in response to a match between the generated and the stored installer identifiers. Once again, *Schneck* neither generates an installer identifier in response to a request to install, nor compares a generated installer identifier to a stored installer identifier, and Figure 11 and column 22, line 51-column 24, line 38 does not disclose that step.

Claim 1 further requires the step of enabling a complete installation of the software product on the local machine, in response to the match between generated and stored installer identifiers. Although the rejection cites column 30, lines 6-47 of *Schneck* as teaching this step, the rejection also acknowledges that *Schneck* does not specifically disclose its teachings may be applied to software installation on a local machine, with software identifiers as particularly claimed for the Applicants' invention. However, *Alexander* is cited as directed to installation of software product onto client machines.

The rejection then concludes that it would have been obvious to one of ordinary skill to combine *Schneck*'s system for controlling access and distribution of digital property with *Alexander*'s software installation techniques, because software is a subset of digital property. The Applicants respectfully disagree that those two references would have led one of ordinary skill to produce their invention as set forth in Claim 1. Although *Alexander* does disclose allowing a user to order, pay for, and unlock a software application, *Schneck* is concerned instead with "controlling access, including preventing or restricting potential use" of data or of the functionality of software. *Schneck* discusses securing components of his data access mechanism 114 (column 15, line 29-column 17, line 33) to prevent users from gaining an authorized access to the data. The entire thrust of *Schneck* is to <u>avoid</u> a complete distribution of a software product on a local machine; only limited access to data, not installation of a software product, is contemplated by *Schneck*. According, one of ordinary skill would find *Alexander*'s teachings of downloading and installing a software application as directly <u>away from</u> the teachings of *Schneck*. For that reason, one of ordinary skill would not have been led to combine those disparate teachings in the manner suggested by the rejection.

For the foregoing reasons, Claim 1 would not have been obvious to one of ordinary skill in view of *Schneck* and *Alexander*.

Claim 11 defines the Applicants' method for restricting installation and execution, on a local machine, of a software product. The method steps recited in Claim 11 are substantially the same as those of Claim 1, discussed above, plus the further step of enabling the execution of the software product on the local machine in response to determining that a proper license file has been stored on the local machine following a match between the generated installer identifier and the stored installer identifier. As discussed above with regard to Claim 1, *Schneck* fails to disclose generating an installer identifier <u>in response to a request to install the software product on the local machine</u>, and also fails to disclose other steps recited in the claimed method. Accordingly, that reference together with *Alexander* would not have made obvious the method of Claim 11, again for the reasons set forth above with respect to Claim 1.

Claims 2-10 and 12-19 depend respectively from Claim 1 and Claim 11, and those dependent claims are patentable over the applied art for the reasons discussed above. Furthermore, Claims 3 and 13 further identify the generated installer identifier as representing a characteristic of the software product. As discussed above, *Schneck* fails to disclose generating an installer identifier in response to a request to install the software product on a local machine, and it follows that *Schneck* does not teach a particular generated installer identifier as set forth in Claims 3 and 13. Likewise, Claims 4-6 and 14-16 further characterize the generated installer identifier and, accordingly, are not taught by *Schneck* and *Alexander*.

Claims 8 and 18 add the step of installing <u>on the local machine</u> at least one run-time file associated with the software product, in response to determining that a received software product key is a correct key. The rejection asserts that columns 18, lines 52-61 and column 34, lines 14-28 of *Schneck* disclose this limitation. However, *Schneck* points out at column 18, line 66-column 19, line 2 that the appropriate rules, if any, are stored within the access mechanism 114 which, as previously disclosed (columns 15 and 16) is a temper-detectable hardware unit. Column 34, lines 14-28 further discuss an access mechanism that may be supplied with the set of rules built-in. These teachings of

Schneck thus are inconsistent with the steps added by Claims 8 and 18, namely, installing at least one run-time file in response to determining that the received software product key is a correct such key.

Independent Claim 20 is rejected as unpatentable over *Schneck* in view of *Alexander*, for the reasons given for the rejection of Claims 1 and 11. The Applicants respectfully traverse the rejection for the reasons discussed above with regard to Claims 1 and 11. In particular, *Schneck* fails to disclose generating an installer identifier representing a characteristic of a software product media containing the software product and comparing the *generated* installer identifier to a *stored* installer identifier on the software product. This step of the Applicants' invention verifies that a media license file was not tempered with prior to installation and that the file was intended to be used with the software product on the media. The Applicants' solution according to Claim 21 thus addresses a problem not germane to *Schneck*'s data access procedure. Accordingly, Claim 21 would not have been obvious in view of that reference and *Alexander*.

The foregoing is submitted as a complete response to the Office Action identified above. The Applicants submit that the present application is in condition for allowance and solicit a notice to that effect.

Respectfully submitted,

MERCHANT & GOULD

Roger T. Frost
Reg. No. 22,176

Date: November 18, 2004

Merchant & Gould, LLC
P.O. Box 2903
Minneapolis, MN 55402-0903
Telephone: 404.954.5100

27488
PATENT TRADEMARK OFFICE